REMARKS/ARGUMENTS

The final Office Action of June 18, 2007 has been carefully reviewed and these remarks are responsive thereto. Claims 1-5, 7-8, 10-39, and 42-45 are pending. Claims 1-5, 7-8, 10-39, and 42-45 are rejected.

Applicant acknowledges that the arguments presented in the paper filed April 24, 2007 with respect to claims 1-5, 7-8, and 10-45 are moot in view of the new grounds of rejection. Applicant acknowledges that the withdrawal of the objection to claim 29.

Reconsideration and allowance of the instant application are respectfully requested.

Claim Rejections – 35 U.S.C. § 102

Claims 1-5, 7-8, 10-39, and 42-45 are rejected under 35 U.S.C. 112, first paragraph, as allegedly failing to comply with the written description.

The Office Action alleges that (Page 2, Section 4.):

> Currently amended claims 1, 38 and 42 recite the new limitations" "autonomously reproduce the capture user event". Examiner was unable to locate any description of the element operation of "autonomously".

Applicant is removing "autonomously" from claims 1, 38, and 42. Applicant requests reconsideration of claims 1-5, 7-8, 10-39, and 42-45.

Claim Rejections – 35 U.S.C. § 102

Claims 1-2, 4-5, 7-8, 12-29, and 32-45 are rejected under 35 U.S.C. 102(e) as allegedly being anticipated by US Publication No. 20040100507 (Hayner).

Regarding claim 1, Applicant is amending the claim to include the feature of "in response to (a), capturing a user event, **from any of a plurality of applications**, associated with the first

screen object." (Emphasis added.) The amendment is supported by the specification as originally

filed. For example, the present specification discloses (Paragraph 24. Emphasis added.):

> Figure 2 shows an exemplary architecture 200 for capturing and processing user
> events (*e.g.*, a user event corresponding to clicking on menu 105 as shown in
> Figure 1) in accordance with an embodiment of the invention. Figure 2 shows an
> exemplary computer system, comprising a user's computer 251 and a help desk's
> computer 253. In the example, shown in Figure 2, a user is manipulating a mouse
> and a keyboard to generate user events that are associated with an application 205.
> **In the embodiment, application 205 is a software program, including a**
> **database manager, spreadsheet, communications package, graphics package,**
> **word processor, and web browser.** The user is operating on desktop 201. For
> example, the user may click or double-click on a screen object (associated with
> application 205) or may enter text into a window corresponding to application
> 205. The user may activate the capturing and processing of user events by
> entering commands through a user interface 207 such as entering a record
> command. (User interface 207 is discussed in more detail with Figure 3.) An
> event engine component 211 receives commands from User Interface so that
> event engine 211 is configured to capture and process user events. In the
> embodiment, event engine 211 is implemented as an ActiveX component that
> may be accessed by a Win32 application as well as by a web page using
> Javascript or a Win32 Visual Basic component. (Event engine 211 is a dynamic
> link library (DLL). In the embodiment, event engine 211 is implemented as an
> ActiveX component, although other embodiments of the invention may
> implement event engine 211 with other software tools and computer languages,
> *e.g.*, Java.) Typical user events include mouse clicks and keystrokes.

One should note that user events may be captured at the operating system level rather than the

application level. Thus, anything that is happening on the operating system from the user may be

recorded.

The Office Action alleges that Hayner discloses (Page 4, section 6.):

> ...(b) in response to (a), capturing a user event associated with the first screen
> object (paragraph 32, 38);

Hayner discloses (Paragraphs 31-32.):

> [0031] Referring now to the drawings, FIG. 1 shows an illustrative system 10 for
> collecting data about user actions. A user interacts with a Web browser 12 being
> operated by a user computational device 14. These interactions may optionally

include, but are not limited to, user input device actions, displaying Web pages
and any type of GUI (graphical user interface) activities. As such, these
interactions preferably include both user actions, such as entering information
through a keyboard and/or "clicking on" or otherwise selecting a GUI gadget
through a pointing device; and reactions to these user actions, such as displaying a
Web page for example. These interactions are preferably collected by a client 16
which is operated by user computational device 14.

[0032] Client 16 is optionally implemented as an applet, such as a Java applet or
ActiveX control (Web browser 12 forms the container); or alternatively as a
software application. The ActiveX control and the software application may both
optionally use the Web browser COM interface or perform hooking on the OS, in
order to capture user action and sniffing on the network layer in order to capture
Web browser sessions.

Referencing fig. 1 of Hayner, Hayner merely discloses a user interacting with only one
application (Web browser 12), in which user actions are collected by client 16 (which may be
implemented as an applet, ActiveX control, or software application). However, Hayner fails to
even suggest the feature of "capturing a user event, from any of a plurality of applications,
associated with the first screen object."

Moreover, Applicant is amending independent claim 38 to include the feature of "a
processing module that captures, from any of a plurality of applications, and processes a user
event by utilizing an application programming interface (API), wherein the user event is
associated with a screen object and wherein the API is coordinate-independent and application
message independent with respect to the screen object." Also, Applicant is amending
independent claim 42 to include the feature of "in response to (a), capturing, from any of a
plurality of applications, a user event associated with the screen object." Claims 2, 4-5, 7-8, 12-
29, 32-37, 39, and 43-45 ultimately depend from claims 1 and 38 and are not anticipated for at
least the above reasons. Applicant requests reconsideration of claims 1-2, 4-5, 7-8, 12-29, and
32-39, and 42-.45.

Also, Applicant is amending claim 15 (which ultimately depends from claim 1) to include the
feature of "adjusting a recording speed associated with the user event based on a recording speed
input, the recording speed being associated with a minimum duration of the user event for

recording the user event." The amendment is supported by the specification as originally filed. For example the specification discloses (Paragraph 36 in reference to Figures 3 and 5.):

> ... In step 505, event engine 211 starts a timer and adjusts a timer speed in accordance with recording speed input 317 (as shown in Figure 3). In step 507, if the left mouse button is depressed for two or more clock iterations, step 509 is executed. Otherwise, step 505 is repeated. ...

The user adjusts the recording speed by changing recording speed input 317. The Office Action alleges that (Page 5.):

> Hayner teaches wherein, determining a speed associated with the user event, the speed being associated with duration of the user event (Paragraph 37 and 46-47; storing timestamp of events associated data for the reconstruction and replaying of the user session.

However, Hayner merely discloses the storing of timestamps and fails to even suggest the feature of "adjusting a recording speed associated with the user event based on a recording speed input, the recording speed being associated with a minimum duration of the user event for recording the user event."

In addition, Applicant is amending claim 45 (which depends from claim 1) to include the feature of "including a note **by the user** to the event entry." (Emphasis added.) The amendment is supported by the specification as originally filed, e.g., Paragraph 31. The Office Action alleges that (Page 7.):

> As per claim 43-45, they are rejected for the same reason as the claims above.

However, Applicant is unable to identify any teaching in Hayner that would suggest the feature of "including a note by the user to the event entry."

Claim Rejections – 35 U.S.C. § 103

Claims 10-11 are rejected under 35 U.S.C. 103(a) as allegedly being unpatentable over Hayner in view of US Patent No. 6,968,509 (Chang).

Claims 10-11 ultimately depend from claim 1. Moreover, Chang fails to remedy the deficiencies of Hayner. Thus, claims 10-11 are patentable for at least the above reasons. Applicant requests reconsideration of claims 10-11.

Moreover, Applicant is amending claim 10 to include the feature of "**modifying a replayed user event by** editing **an attribute** of the event entry of the file." (Emphasis added.) The amendment is supported by the specification as originally filed, e.g., Paragraph 31. The Office Action alleges that (Page 8. Emphasis added.):

> Hayner does **not** teach, editing the event entry of a file. However in the same field of endeavor, recording of user events Chang discloses edit menu item and event entries (606) in (fig. 6-10).

However, Chang fails to suggest anything about editing an attribute of an event entry. For example, as shown in the screenshots in figs. 7 and 8 of Chang, the user is performing user events as the user is editing when using word processor 126. (Column 6, line 61-column 7, line 19.) Chang merely discloses recording a user event in which the user is editing text, but Chang does not even suggest editing attributes of the event entry.

Claims 30-31 are rejected under 35 U.S.C. 103(a) as allegedly being unpatentable over Hayner in view of US Patent No. 6,662,226 (Wang).

Claims 30-31 ultimately depend from claim 1. Moreover, Wang fails to remedy the deficiencies of Hayner. Thus, claims 30-31 are patentable for at least the above reasons. Applicant requests reconsideration of claims 30-31.

Moreover, claim 30 includes the feature of "in response to (i), **drilling down through a hierarchy** to find a matching screen object in accordance with at least one attribute of the event entry." (Emphasis added.) The Office Action alleges (Page 9. Emphasis added.):

> Hayner does not clearly discloses enumerating a desktop, drilling down through hierarchy to find a matching screen object, stopping playback of the file, invoking a recorded action that is associated with the user event, proceeding to a next user event that is recorded by the file. However, Wang discloses a system method and product that typically permit a user/administrator to retrieve, track and replay recorded screen activities; drilling down to find a matching screen, tracking playback of file, invoking actions and preceding to a next user event (Figure 416B-C; col. 9, lines 18-67).

Wang discloses (Column 9, lines 18-33. Emphasis added.):

> FIG. 6B shows a process 640 of a server communicating with the terminal device as discussed in FIG. 6A. At 642, the server awaits a request if any file is upcoming from a terminal device such as the one in FIG. 6A. **When a request is received, the server assigns an identification to the file so that a query can be made later to retrieve the file when there is a need to review the file.** Depending on an exact implementation or application, the identification may be a session ID, a transaction ID or any ID that can uniquely identify the file. At 646, the file is received. **At 648, it is determined if there is another file related to the arrived file. As described above, sometimes, there are multiple files each comes individually for one transaction and sometime there is only one compounded file.** If it is determined that there are no more files, the arrived files are then kept in a storage space at 650 for future retrieval.

However, Wang merely discloses assigning a transaction identification (step 644 as shown in fig. 2) that merely identifies the file. (Column 9, lines 24-33.) As disclosed in the above teaching, the transaction identification is used for determining whether there are multiple files for a transaction. However, Wang fails to even suggest a <u>hierarchical</u> relationship among files.

All rejections having been addressed, applicants respectfully submit that the instant application is in condition for allowance, and respectfully solicit prompt notification of the same.

Respectfully submitted,

BANNER & WITCOFF, LTD.

Dated: November 16, 2007   By: _Kenneth F. Smolik_

Kenneth F. Smolik
Registration No. 44,344
BANNER & WITCOFF, LTD.
10 S. Wacker Drive, Suite 3000
Chicago, IL 60606-7407
Telephone: 312-463-5419
Facsimile: 312-463-5001